to use metadata to define schemas for a datamart. From the definition of the schema, the system can automatically generate the tables in the datamart. Further, the system can automatically extract the data from the source systems, perform conversions on that data and populate the datamart. The system supports the automatic creation and processing of aggregates from aggregate definitions. The system also supports the creation of the query mechanisms from query definitions.

Please replace the paragraph beginning at page 15, line 17 with the following amended paragraph:

The measurement information 168 and the query/reporting information 169 support the querying of the datamart 150. A measure is a piece of numeric data in the datamart 150 that is useful to a user. That is, individual fact columns from source systems can be very implementation specific. These columns may not correspond to what users would prefer to see. For example, a user may want to see a net price added with a total cost. However, the fact table may only include the net price of the total cost. The measurement information 168 allows the consultant to define the abstract notation of the calculation associated with the net price added to the total cost.

Please replace the paragraph beginning at page 28, line 4 with the following amended paragraph:

The dimension base 306 is the metadata 160 describing all the dimension tables that can be used in a given constellation 302. These dimension bases can then be used in multiple constellations. The dimension base 306 includes the following attributes: an aggregate key operator, a cleanse flag, a description, a dimension base key, dimension base name, a dimension base type, and a truncate stage flag. The aggregate key operator is an SQL operator used by the aggregate builder 170 to build aggregates from a dimension. The cleanse flag and description act similarly to those attributes in other tables. The dimension base key is the primary key for the dimension base 306. The dimension base name is the name of the base dimension used in constructing real tables in the datamart 150. The dimension base type indicates the type of a dimension base, either default or special (special includes "date" and "transaction type," which are used by the system 100). The truncate stage flag operates in the manner similar to other truncate stage flags.

LAW OFFICES OF
SKJERVEN MORRILL
MACPHERSON LLP

25 METRO DRIVE
SUITE 700
SAN JOSE, CA 95110
(408) 453-9300

Please replace the paragraph beginning at page 39, line 3 with the following amended paragraph:

The job log 401 is the location where the running job is logged. That is, the location of the output that will be provided to the consultant indicating what occurred during the extraction (e.g., what errors occurred). The job log 401 includes a data key, a job key, a job log key, and a job store role. The data store key indicates the data store having a role defined within the job. The job key is a reference to the particular job, the job log key is the primary key for the job log 401. The job store role is the role being assigned to that particular job log 401. An example job store is "<working directory>", indicating the path to the working directory where job log files are stored.

Please replace the paragraph beginning at page 47, line 16 with the following amended paragraph:

The SQL server store table 456 defines details about an SQL server system. The SQL server store includes the following attributes: a data store key, a database name, a password, a server, an SQL server store key, a user name, and a version. The data store key is a one to one relationship to a data store entry. The database name is an SQL server database name ($$DEFAULT means the database in which this role resides). The password is the SQL server password. $$DEFAULT again means the password currently logged into to read this data. The server is the SQL server name. The SQL store key is the primary key. The user name is the SQL server user name. The version is the vendor's version number of this SQL server. $$DEFAULT means use the default value for the current database being used. For example, the database name means the database in which this role resides.

Please replace the paragraph beginning at page 48, line 13 with the following amended paragraph:

The job defines the order of the execution of the connectors. The job also allows for the running of an external program, such as system call, between connector executions. Thus, each job step in a job can be a system call or a connector.

Serial No. 09/073,748

Please replace the paragraph beginning at page 48, line 16 with the following amended paragraph:

The following is an example illustrating the organization of a job. Assume that a consultant wants to extract information from a source system that provides a raw set of home addresses. A system call could be run as part of a job step. The system call would determine the zip codes associated with those addresses. The zip codes could then be included in the datamart 150.

Please replace the paragraph beginning at page 52, line 18 with the following amended paragraph:

The team template is used to properly populate an "associative" dimension table. Such a table is used whenever there is a one-to-many relationship between an individual fact row and a dimension. For example, if multiple salespeople can split the credit for an order, one needs some way to represent this situation in the datamart. In a star schema, one normally associates a tuple of dimension values with a fact row (e.g. product, customer, salesrep dimensions for the fact row containing price, quantity etc.). Since there is only a single salesrep_key, one could normally have only one salesrep associated with this transaction. There are two solutions. One is to introduce multiple fact rows for a transaction involving one to many relationships. If there were three salesreps on a specific order, there would be three fact rows for this order stored in the database. This has the disadvantage of multiplying the data size by a factor of three and slows queries. Also queries that are concerned with the total number of transactions become more difficult to process since duplicate rows, due to the multiplication by the number of salesreps, must be eliminated.

Please replace the paragraph beginning at page 59, line 13 with the following amended paragraph:

The actual column type 540 is a logical description of the role a column plays in the system 100. The actual column type 540 includes attributes that define value to be used in a database for a column of this type.

Please replace the paragraph beginning at page 62, line 6 with the following amended paragraph:

LAW OFFICES OF
SKJERVEN MORRILL
MACPHERSON LLP

25 METRO DRIVE
SUITE 700
SAN JOSE, CA 95110
(408) 453-9200

728336 v1 / PF-OA [Rev. 000913]

-4-   Serial No. 09/073,748

In some embodiments of the system, the aggregate tables are used to answer queries about backlog/balance/inventory quantities. In order to answer such queries, the previously described rolling forward from the beginning of time is done. However, this is performed efficiently through the accessing of the appropriate time aggregates. For example, assume the datamart 150 has five years of historical transaction data beginning in 1993. Assume that one desires the inventory of some specified products on May 10, 1996. This would be computed by querying all of the transactions in the 1993, 1994 and 1995 year aggregates, the 1996 Q1 quarter aggregate, the April 1996 month aggregate, the May 1996 week 1 aggregate and finally 3 days of actual May, 1996 daily transactions. These transactions (additions and subtractions from inventory) would be added to the known starting inventory in order to produce the inventory on May 10. Note that this time navigation "hops" by successively smaller time intervals (year, quarter, month, week, day) in order to minimize the number of database accesses. What is important is the exploitation of aggregate tables that already exist in the system in order to answer transactional queries rapidly (e.g. What were the total sales of product X in Apr 1996?). This avoids the need to build what is essentially a second datamart with the balance/inventory/backlog snapshots.

Please replace the paragraph beginning at page 68, line 19 with the following amended paragraph:

The following describes the filtering tables. The filter block 650 is a top level grouping table for filter area within a ticksheet. The filter block 659 is tied to a particular dimension column in the schema definition. The filter block 650 includes columns, a description, a dictionary key, a dimension column key, a dimension role key, a filter block key, a filter block type a label, a list order, a name, a plural, a mapping flag, and a ticksheet key. The columns field indicates the number of columns in this filter block. The description is for documentation. The dictionary key points to the help dictionary. The dimension column key points to the actual column name and the datamart to be filtered on. A null value here means degenerate dimension as determined by the dimension role key. The dimension role key points to the dimension role in the constellation of the ticksheet that is the form key to filter on for all facts in this constellation. A null value here means that a special dimension shared by all constellations is being used. The filter block key is the primary key for this table. The filter block type points to the filter block type table 652 which defines the ways in

728336 v1 / PF-OA [Rev. 000913]

Serial No. 09/073,748